In [9]:
```
# 9-1 & 9-2
class Restaurant():
    def __init__(self, restaurant_name, restaurant_type):
        self.restaurant_name = restaurant_name
        self.restaurant_type = restaurant_type

    def describe_restaurant(self):
        print(self.restaurant_name + " is a " + self.restaurant_type + " restaurar


    def open_restaurant(self):
        print(self.restaurant_name + " is open.")

my_restaurant = Restaurant('dongtingchun', 'chinese')
my_restaurant.describe_restaurant()
my_restaurant.open_restaurant()

print("\n")
your_restaurant = Restaurant('stone', 'korean')
your_restaurant.describe_restaurant()
your_restaurant.open_restaurant()

print("\n")
her_restaurant = Restaurant('homegrown', 'american')
her_restaurant.describe_restaurant()
her_restaurant.open_restaurant()
```

```
dongtingchun is a chinese restaurant.
dongtingchun is open.


stone is a korean restaurant.
stone is open.


homegrown is a american restaurant.
homegrown is open.
```

In [11]:
```
# 9-3

class user():
    def __init__(self, first_name, last_name):
        self.first_name = first_name
        self.last_name = last_name

    def user_name(self):
        print("Current user is " + self.first_name + " " + self.last_name)


    def greeting(self):
        print("hello " + self.first_name + " " + self.last_name + "!")

user1 = user('Wei', 'Wang')
user1.user_name()
user1.greeting()

print("\n")
user2 = user('Xiao', 'Wu')
user2.user_name()
user2.greeting()

print("\n")
user3 = user('Kong', 'Wu')
user3.user_name()
user3.greeting()
```

```
Current user is Wei Wang
hello Wei Wang!


Current user is Xiao Wu
hello Xiao Wu!


Current user is Kong Wu
hello Kong Wu!
```

In [33]:
```
# 9-4
class Restaurant():
    def __init__(self, restaurant_name, restaurant_type):
        self.restaurant_name = restaurant_name
        self.restaurant_type = restaurant_type
        self.number_served = 0

    def describe_restaurant(self):
        print(self.restaurant_name + " is a " + self.restaurant_type + " restaurar

    def served(self):
        print("Today it served " + str(self.number_served) + " people.")


    def open_restaurant(self):
        print(self.restaurant_name + " is open.")

    def number_served(self, number):
        self.served = number

my_restaurant = Restaurant('dongtingchun', 'chinese')
my_restaurant.describe_restaurant()
my_restaurant.open_restaurant()
my_restaurant.number_served = 12
my_restaurant.served()
```

```
dongtingchun is a chinese restaurant.
dongtingchun is open.
Today it served 12 people.
```

In [21]:
```
# 9-4
class Restaurant():
    def __init__(self, restaurant_name, restaurant_type):
        self.restaurant_name = restaurant_name
        self.restaurant_type = restaurant_type
        self.number_served = 0

    def describe_restaurant(self):
        print(self.restaurant_name + " is a " + self.restaurant_type + " restaurar

    def served(self):
        print("Today it served " + str(self.number_served) + " people.")


    def open_restaurant(self):
        print(self.restaurant_name + " is open.")

my_restaurant = Restaurant('dongtingchun', 'chinese')
my_restaurant.describe_restaurant()
my_restaurant.open_restaurant()
my_restaurant.served()
```

```
dongtingchun is a chinese restaurant.
dongtingchun is open.
Today it served 0 people.
```

In [35]:
```
# 9-4
class Restaurant():
    def __init__(self, restaurant_name, restaurant_type):
        self.restaurant_name = restaurant_name
        self.restaurant_type = restaurant_type
        self.number_served = 0

    def describe_restaurant(self):
        print(self.restaurant_name + " is a " + self.restaurant_type + " restaurar

    def served(self):
        print("Today it served " + str(self.number_served) + " people.")


    def open_restaurant(self):
        print(self.restaurant_name + " is open.")

    def new_number(self, number):
        self.number_served = number

my_restaurant = Restaurant('dongtingchun', 'chinese')
my_restaurant.describe_restaurant()
my_restaurant.open_restaurant()
my_restaurant.new_number(12)
my_restaurant.served()
```

```
dongtingchun is a chinese restaurant.
dongtingchun is open.
Today it served 12 people.
```

In [38]:
```
# 9-5

class user():
    def __init__(self, first_name, last_name):
        self.first_name = first_name
        self.last_name = last_name
        self.login_attempts = 0

    def user_name(self):
        print("Current user is " + self.first_name + " " + self.last_name)


    def greeting(self):
        print("hello " + self.first_name + " " + self.last_name + "!")

    def increment_login_attempts(self):
        attempts = self.login_attempts += 1

    def attempts(self):
        print("You have logined " + str(self.login_attempts) + " times.")

user1 = user('Wei', 'Wang')
user1.user_name()
user1.greeting()
user1.attempts()
```

```
Current user is Wei Wang
hello Wei Wang!
You have logined 0 times.
```

In [3]:
```
# 9-5

class user():
    def __init__(self, first_name, last_name):
        self.first_name = first_name
        self.last_name = last_name
        self.login_attempts = 0

    def user_name(self):
        print("Current user is " + self.first_name + " " + self.last_name)


    def greeting(self):
        print("hello " + self.first_name + " " + self.last_name + "!")


    def increment_login_attempts(self):
       self.login_attempts += 1

    def attempts(self):
        print("You have logined " + str(self.login_attempts) + " times.")

    def reset_login_attempts(self):
        self.login_attempts = 0

user1 = user('Wei', 'Wang')
user1.user_name()
user1.greeting()

login_count = 0
while login_count < 10:
    user1.increment_login_attempts()
    user1.attempts()
    login_count += 1
user1.reset_login_attempts()
user1.attempts()
```

```
Current user is Wei Wang
hello Wei Wang!
You have logined 1 times.
You have logined 2 times.
You have logined 3 times.
You have logined 4 times.
You have logined 5 times.
You have logined 6 times.
You have logined 7 times.
You have logined 8 times.
You have logined 9 times.
You have logined 10 times.
You have logined 0 times.
```

In [10]:
```
# 9-6

class Restaurant():
    def __init__(self, restaurant_name, restaurant_type):
        self.restaurant_name = restaurant_name
        self.restaurant_type = restaurant_type
        self.number_served = 0

    def describe_restaurant(self):
        print(self.restaurant_name + " is a " + self.restaurant_type + " restaurar

    def served(self):
        print("Today it served " + str(self.number_served) + " people.")


    def open_restaurant(self):
        print(self.restaurant_name + " is open.")

    def number_served(self, number):
        self.served = number

class IceCreamStand(Restaurant):
    def __init__(self, restaurant_name, restaurant_type, flavor):
        super().__init__(restaurant_name, restaurant_type)
        self.flavor = flavor

    def describe_flavor(self):
        print("The most popular flavor is " + self.flavor)

one_IceCreamStand = IceCreamStand('coldstone', 'icecream', 'cheesecake')
one_IceCreamStand.describe_restaurant()
one_IceCreamStand.open_restaurant()
one_IceCreamStand.describe_flavor()
```

```
coldstone is a icecream restaurant.
coldstone is open.
The most popular flavor is cheesecake
```

In [15]:
```
# 9-7
class user():
    def __init__(self, first_name, last_name):
        self.first_name = first_name
        self.last_name = last_name

    def user_name(self):
        print("Current user is " + self.first_name + " " + self.last_name)


    def greeting(self):
        print("hello " + self.first_name + " " + self.last_name + "!")

class admin(user):
    def __init__(self, first_name, last_name):
        super().__init__(first_name, last_name)
        self.privileges = "can delete post"

    def show_privileges(self):
        print(self.first_name + " " + self.privileges)

admin1 = admin('Wei', 'Wang')
admin1.user_name()
admin1.greeting()
admin1.show_privileges()
```

```
Current user is Wei Wang
hello Wei Wang!
Wei can delete post
```

In [41]:
```
class user():
    def __init__(self, first_name, last_name):
        self.first_name = first_name
        self.last_name = last_name

    def user_name(self):
        print("Current user is " + self.first_name + " " + self.last_name)


    def greeting(self):
        print("hello " + self.first_name + " " + self.last_name + "!")

class admin(user):
    def __init__(self, first_name, last_name, privileges=["can add post", "can del
        super().__init__(first_name, last_name)
        self.admin_privilege = Privileges(privileges)

    def show_privileges(self):
        print(self.first_name + " privileges:")
        print(self.admin_privilege.get_privilieges())


class Privileges():
    privilieges = ["can add post", "can delete post", "can ban user"]

    def __init__(self, privileges):
        self.privilieges = privileges.copy()


    def get_privilieges(self):
        return self.privilieges


user1 = admin('Wei', 'Wang', ["can add post", "can delete post"])
user1.show_privileges()

user2 = admin('Xiao', 'Wu')
user2.show_privileges()
```

```
Wei privileges:
['can add post', 'can delete post']
Xiao privileges:
['can add post', 'can delete post', 'can ban user']
```

In [ ]:
```
# 9-9
```

In [50]:
```
class Car():
    def __init__(self, make, model, year):
        self.make = make
        self.model = model
        self.year = year
        self.odometer_reading = 0

    def get_descriptive_name(self):
        long_name = str(self.year) + ' ' + self.make + ' ' + self.model
        return long_name.title()

    def read_odometer(self):
        print("This car has " + str(self.odometer_reading) + " miles on it.")

    def update_odometer(self, mileage):
        if mileage >= self.odometer_reading:
            self.odometer_reading = mileage
        else:
            print("You can't roll back an odometer!")

    def increment_odometer(self, miles):
        self.odometer_reading += miles


class Battery():

    def __init__(self, battery_size=70):
        self.battery_size = battery_size

    def describe_battery(self):
        print("This car has a " + str(self.battery_size) + "-kWh battery.")

    def get_range(self):
        if self.battery_size == 70:
            range = 240
        elif self.battery_size == 85:
            range = 270
        message = "This car can go approximately " + str(range)
        message += " miles on a full charge."
        print(message)


    def upgrade(self):
        if self.battery_size != 85:
            self.battery_size = 85



class ElectricCar(Car):
    def __init__(self, make, model, year):
        super().__init__(make, model, year)
        self.battery = Battery()

    def get_range(self):
        if self.battery_size == 70:
            range = 240
```

```
        elif self.battery_size == 85:
            range = 270




my_tesla = ElectricCar('tesla', 'model s', 2016)
print(my_tesla.get_descriptive_name())
my_tesla.battery.describe_battery()
my_tesla.battery.get_range()
my_tesla.battery.upgrade()
my_tesla.battery.get_range()
```

```
2016 Tesla Model S
This car has a 70-kWh battery.
This car can go approximately 240 miles on a full charge.
This car can go approximately 270 miles on a full charge.
```

In [54]:
```
# 9-10
import restaurant

restaurant2 = Restaurant('pho', 'soup')
restaurant2.describe_restaurant()
restaurant2.open_restaurant()
```

```
pho is a soup restaurant.
pho is open.
```

In [55]:
```
# 9-11
from admin import admin, Privileges
user3 = admin('kong', 'Wu')
user3.show_privileges()
```

```
Wei privileges:
['can add post', 'can delete post']
Xiao privileges:
['can add post', 'can delete post', 'can ban user']
kong privileges:
['can add post', 'can delete post', 'can ban user']
```

In [74]:
```
# 9-12
import user
from amdin_privileges import admin

user4 = admin('kong', 'Wu')
user4.show_privileges()
```

```
kong privileges:
['can add post', 'can delete post', 'can ban user']
```

In [76]:
```
# 9-13
from collections import OrderedDict

Glossary = OrderedDict()

Glossary['str'] = 'string'
Glossary['int'] = 'integar'
Glossary['len'] = 'length'
Glossary['del'] = 'delete'

for code, meaning in Glossary.items():
    print(code + ": " + meaning)
```

```
str: string
int: integar
len: length
del: delete
```

In [81]:
```
from random import randint
class Die():
    side = 0
    def __init__(self, side=6):
        self.side = side
    def roll_die(self):
        return randint(1, self.side)

die10 = Die(10)
die20 = Die(20)

i = 0
while i < 10:
    print ("10 side die, i = " + str(i) + ", roll_die = " + str(die10.roll_die()))
    i += 1

i = 0
while i < 10:
    print ("20 side die, i = " + str(i) + ", roll_die = " + str(die20.roll_die()))
    i += 1
```

```
10 side die, i = 0, roll_die = 5
10 side die, i = 1, roll_die = 10
10 side die, i = 2, roll_die = 4
10 side die, i = 3, roll_die = 7
10 side die, i = 4, roll_die = 5
10 side die, i = 5, roll_die = 7
10 side die, i = 6, roll_die = 4
10 side die, i = 7, roll_die = 8
10 side die, i = 8, roll_die = 4
10 side die, i = 9, roll_die = 2
20 side die, i = 0, roll_die = 12
20 side die, i = 1, roll_die = 17
20 side die, i = 2, roll_die = 5
20 side die, i = 3, roll_die = 18
20 side die, i = 4, roll_die = 8
20 side die, i = 5, roll_die = 11
20 side die, i = 6, roll_die = 19
20 side die, i = 7, roll_die = 19
20 side die, i = 8, roll_die = 14
20 side die, i = 9, roll_die = 1
```

In [85]:
```
# 10-1
with open('learning.txt') as file_object:
    contents = file_object.read()
    print(contents)

print("\n")
with open('learning.txt') as file_object:
    contents = file_object.read()
    print(contents.rstrip())

print("\n")
filename = 'learning.txt'
with open(filename) as file_object:
    for line in file_object:
        print(line)
```

```
In Python you can creat list and dictionary.
In python you can define functions.


In Python you can creat list and dictionary.
In python you can define functions.


In Python you can creat list and dictionary.

In python you can define functions.
```

In [91]:
```
# 10-2
with open('learning.txt') as file_object:
    contents = file_object.read()

    replaced = contents.replace('Python', 'C')

    print(replaced)
```

```
In C you can creat list and dictionary.
In python you can define functions.
```

In [1]:
```
# 10-3
filename = 'guest.txt'
with open(filename, 'w') as file_object:
    while True:
        file_object.write(input("Please write down your name: ") + "\n")
        file_object.write(input("Please write down your name: "))
        break
```

```
Please write down your name: Wei Wang
Please write down your name: Xiao Wu
```

In [2]:
```
# 10-4
filename = 'guest_book.txt'
with open(filename, 'w') as file_object:
    while True:
        message = input("Please write down your name: ")
        file_object.write(message + "\n")
        print("Hello, " + message)
        message = input("Please write down your name: ")
        file_object.write(message + "\n")
        print("Hello, " + message)
        break
```

```
Please write down your name: Wei Wang
Hello, Wei Wang
Please write down your name: Xiao Wu
Hello, Xiao Wu
```

In [3]:
```
# 10-5
filename = 'reason.txt'
with open(filename, 'w') as file_object:
    while True:
        file_object.write(input("Please write down the reason why you like Python:
        file_object.write(input("Please write down the reason why you like Python:
        break
```

```
Please write down the reason why you like Python: It's efficient.
Please write down the reason why you like Python: It's funny.
```

In [8]:
```
# 10-6
first_number = input("Please enter a nunmer: ")
second_number = input("Please enter a nunmer one more time: ")
try:
    summary = int(first_number ) + int(second_number)
    print(summary)
except ValueError:
    print("You need to enter a number.")
```

```
Please enter a nunmer: 5
Please enter a nunmer one more time: a
You need to enter a number.
```

In [2]:
```
# 10-7

while True:
    first_number = input("Please enter a number: ")
    second_number = input("Please enter a number one more time: ")
    if first_number == 'quit' or second_number == 'quit':
        break
    else:
        try:
            summary = int(first_number ) + int(second_number)
            print(summary)
        except ValueError:
            print("You need to enter a number.")
```

```
Please enter a number: 9
Please enter a number one more time: 1
10
Please enter a number: w
Please enter a number one more time: 7
You need to enter a number.
Please enter a number: 5
Please enter a number one more time: 3
8
Please enter a number: quit
Please enter a number one more time: quit
```

In [3]:
```
# 10-8
filename = 'cats.txt'
try:
    with open(filename) as file_object:
        contents = file_object.read()
        print(contents)
except FileNotFoundError:
    print("Sorry, the file" + filenmae + " does not exist.")

filename = 'dogs.txt'
try:
    with open(filename) as file_object:
        contents = file_object.read()
        print(contents)
except FileNotFoundError:
    print("Sorry, the file" + filenmae + " does not exist.")

filename = 'dogs.txt'
```

```
Alice
Bill
Cathy
Dan
Emma
Fannie
```

In [7]:
```
# 10-8
filename = 'cats.txt'
try:
    with open(filename) as file_object:
        contents = file_object.read()
        print(contents)
except FileNotFoundError:
    print("Sorry, the file" + filenmae + " does not exist.")

filename = 'dogs.txt'
try:
    with open(filename) as file_object:
        contents = file_object.read()
        print(contents)
except FileNotFoundError:
    print("Sorry, the file " + filename + " does not exist.")
```

```
Alice
Bill
Cathy
Sorry, the file dogs.txt does not exist.
```

In [8]:
```
# 10-9
filename = 'cats.txt'
try:
    with open(filename) as file_object:
        contents = file_object.read()
        print(contents)
except FileNotFoundError:
    print("Sorry, the file" + filenmae + " does not exist.")

filename = 'dogs.txt'
try:
    with open(filename) as file_object:
        contents = file_object.read()
        print(contents)
except FileNotFoundError:
    pass
```

```
Alice
Bill
Cathy
```

In [11]:
```
# 10-10
filename = 'drama.txt'
try:
    with open(filename) as file_object:
        contents = file_object.read()

except FileNotFoundError:
    print("Sorry, the file" + filenmae + " does not exist.")

else:
    words = contents.split()
    num_words = len(words)
    print(filename + " has about " + str(num_words) + " words.")

    contents.lower().count('the')
```

```
drama.txt has about 28169 words.
```

In [3]:
```
# 10-11
import json
def get_favorate_number():
    while True:
        str = input("Please enter your favorate number: ")
        try:
            return int(str)
        except ValueError:
            print("That's not an int! Please retry")

def save_favorate_number(number):
    filename = 'number.json'
    with open(filename, 'w') as f_obj:
        json.dump(number, f_obj)

def load_favorate_number():
    filename = 'number.json'
    with open(filename, 'r') as f_obj:
        saved_number = json.load(f_obj)
    return saved_number


number = get_favorate_number()
save_favorate_number(number)
number = load_favorate_number()

print("I know your favorite number! It’s " + str(load_favorate_number()) + "!")
```

```
Please enter your favorate number: sdfsf
That's not an int! Please retry
Please enter your favorate number: safs
That's not an int! Please retry
Please enter your favorate number: 123
I know your favorite number! It’s 123!
```

In [2]:
```
# 10-12
import json

def get_favorate_number():
    while True:
        str = input("Please enter your favorate number: ")
        try:
            return int(str)
        except ValueError:
            print("That's not an int! Please retry")

def save_favorate_number(number):
    filename = 'number10-12.json'
    with open(filename, 'w') as f_obj:
        json.dump(number, f_obj)

def load_favorate_number():
    try:
        filename = 'number10-12.json'
        with open(filename, 'r') as f_obj:
            saved_number = json.load(f_obj)
        return saved_number
    except:
        number = get_favorate_number()
        save_favorate_number(number)
        return number



print("I know your favorite number! It’s " + str(load_favorate_number()) + "!")
```

```
I know your favorite number! It’s 123!
```

In [17]:
```
#10-13

import json

def get_stored_username():
    """Get stored username if available."""
    filename = 'username11-13.json'
    try:
        with open(filename) as f_obj:
            username = json.load(f_obj)
    except FileNotFoundError:
        return None
    else:
        return username

def get_new_username():
    """Prompt for a new username."""
    username = input("What is your name? ")
    filename = 'username11-13.json'
    with open(filename, 'w') as f_obj:
        json.dump(username, f_obj)
    return username

def greet_user():
    """Greet the user by name."""
    username = get_stored_username()
    if username:
        print("Welcome back, " + username + "!")
    else:
        username = get_new_username()

def get_favorate_number():
    while True:
        str = input("Please enter your favorate number: ")
        try:
            return int(str)
        except ValueError:
            print("That's not an int! Please retry")

def save_favorate_number(number):
    filename = 'number10-13.json'
    with open(filename, 'w') as f_obj:
        json.dump(number, f_obj)

def load_favorate_number():
    filename = 'number10-13.json'
    with open(filename, 'r') as f_obj:
        saved_number = json.load(f_obj)
    return saved_number

saved_username = get_stored_username()
username = get_new_username()
if saved_username == username:
    greet_user()
    print("We remember your fav number, " + str(load_favorate_number()) + "!")
else:
```

```
number = get_favorate_number()
save_favorate_number(number)
print("We will remember your fav number next time, " + username + "!")
```

```
What is your name? Xiao Wu
Please enter your favorate number: 456
We will remember your fav number next time, Xiao Wu!
```

In [ ]:

In [ ]: